Generalized Additive Models

Marina Peñalver Ripoll

Contents

Introduction

Generalized Additive Models (GAM) are an advanced and flexible extension of generalized linear models. These models are particularly useful in situations where the relationship between predictors and the response variable is unknown or suspected to be nonlinear. Unlike generalized linear models, GAMs allow the response variable to follow various distributions from the exponential family, such as binomial, Poisson, or normal, making them suitable for a wide variety of data types and analytical contexts.

The structure of a GAM is

$$g(\mu_i) = \mathbf{X}_i^* \theta + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}), \tag{1}$$

where $\mu_i = \mathrm{E}(Y_i)$ and the variables $Y_1, \ldots, Y_n$ follow the same distribution from the exponential family. The function g is a known, twice differentiable (link function); $\mathbf{X}_i^*$ is the i-th row of the strictly parametric model matrix, with associated parameter vector θ; the f_j are smooth functions of variables X_j.

Like in additive models, estimating GAMs requires careful specification of the bases of the smooth functions and a clear definition of what is considered as the smoothness of these functions. For each smooth function $f_j(x_j)$ to be included in the model, a basis must be selected, denoted as $(b_{j1}(x), ..., b_{jq_j}(x))$. The function $f_j(x_j)$ is represented as a linear combination of these bases, as indicated in the following expression:

$$f_j(x_{ij}) = \sum_{k=1}^{q_j} \beta_{jk} b_{jk}(x_{ij}), \tag{2}$$

where β_{jk} are the coefficients associated with each basis, which need to be estimated during the model fitting. The choice of q_j, the number of bases for the j-th function, is crucial and must be adequate to capture the complexity and variability of the relationships between variables.

A GAM can be structured in the form of a penalized GLM. This approach allows the application of techniques already studied in the context of GLMs for estimation and penalization, facilitating computational implementation and analysis. The steps to rewrite a GAM in this form are described below.

First, for each function $f_j(x_j)$, define the matrix $\tilde{\mathbf{X}}^j$, of dimensions $(n \times q_j)$, corresponding to each vector x_j that includes the n observed values of the variable X_j, as $\tilde{\mathbf{X}}^j_{ik} = b_{jk}(x_{ij})$. Define the vector $\beta_j = [\beta_{j1}, \beta_{j2}, \ldots, \beta_{jq_j}]$ of the parameters for each function f_j. Thus, the smooth function for each predictor x_{ij} can be expressed as

$$f_j(x_{ij}) = \tilde{\mathbf{X}}^j \beta_j.$$

By integrating all the smooth functions into the model, an extended matrix $\boldsymbol{X} = [\mathbf{X}^*, \tilde{\mathbf{X}}^1, \tilde{\mathbf{X}}^2, \ldots, \tilde{\mathbf{X}}^p]$ and an extended coefficient vector $\boldsymbol{\beta} = [\theta^T, \beta_1^T, \ldots, \beta_p^T]$ are constructed.

With these definitions, the GAM model can be rewritten in terms of a GLM as follows:

$$g(\mu_i) = \boldsymbol{X}_i\boldsymbol{\beta}. \tag{3}$$

R implementation

`gam()`

The `gam()` function is designed to fit a generalized additive model, as well as other generalized linear models that require quadratic penalties.

During the fitting process, the smoothing degree of the model terms is estimated. These smoothed terms are represented using penalized regression splines or similar methods, selecting the smoothing parameters using criteria such as GCV (Generalized Cross Validation), UBRE (Un-Biased Risk Estimator), AIC (Akaike Information Criterion), REML (Restricted Maximum Likelihood) or NCV (Non-Negative Cross-Validation). Alternatively, splines with fixed degrees of freedom can be used.

The usage of the `gam()` function is as follows:

```
gam(formula, family=gaussian(), data=list(), weights=NULL, subset=NULL,
na.action, offset=NULL, method="GCV.Cp", optimizer=c("outer", "newton"),
control=list(), scale=0, select=FALSE, knots=NULL, sp=NULL, min.sp=NULL,
H=NULL, gamma=1, fit=TRUE, paraPen=NULL, G=NULL, in.out,
drop.unused.levels=TRUE, drop.intercept=NULL, nei=NULL, discrete=FALSE, ...)
```

The key parameters are:

- **`formula`**: Defines the structure of the model. It incorporates both linear and non-linear effects through terms such as `s()` for individual smoothers and `te()` for smooth interactions between multiple variables.
- **`family`**: Defines the distribution family of the model. Common options include `gaussian`, `binomial`, `poisson`, among others.
- **`data`**: The dataset used, which must contain all variables specified in the formula.
- **`method`**: Method for estimating the smoothing parameters, including options like GCV, UBRE, and REML.
- **`knots`**: Specifies the knots for the splines in smoothed terms.

For very large datasets, it is suggested to use the `bam` function, and for mixed GAM models that include random effects, use `gamm`.

`anova()`

The `anova()` function is designed to facilitate the analysis of variance for fitted generalized additive models (GAM). It also allows for meaningful comparisons between different GAM models to assess which one provides the best fit to the data under consideration.

This function is primarily used to compare the fit of GAM models that differ in their terms, to determine if the inclusion of additional terms in the model significantly improves the fit. Additionally, it can be used to test the significance of specific terms within a model, providing a rigorous method for feature selection.

The key parameters are:

- **`object`**: A GAM model or a list of GAM models to be compared.
- **`test = c("Chisq", "F", "Cp")`**: Specifies the type of statistical test to use for the comparison.

`gam.check()`

The `gam.check()` function is an essential tool for checking and diagnosing GAM models. This function is used to evaluate the adequacy of the model fit and to detect potential issues that could affect the interpretation and validity of the model.

The function performs a series of tests and diagnostic plots that help assess the quality of the model fit. It offers four forms of visualization, which are:

1. **Residual Plots**: Provides residual plots to check for homogeneity of variance and independence of residuals.
2. **Residual Histogram**: Displays a histogram of the standardized residuals to assess their distribution. An approximately normal distribution of residuals is a good indication that the model is appropriate.
3. **Q-Q Plot of Residuals**: A quantile-quantile plot that helps check the normality of the residuals, which is important for many statistical inferences in the context of GAMs.
4. **Scatter Plot of Predicted vs. Observed Responses**: Helps assess linearity and the overall fit of the model.

Additionally, it performs tests for each smoothed term in the model to determine if the level of smoothing is appropriate. This includes checking if the number of degrees of freedom is sufficient to capture the underlying relationship without overfitting.